Visualization of Conway Polyhedron Notation

Hidetoshi Nonaka

***Abstract*—**This paper presents an interactive modeling system of polyhedra using the isomorphic graphs. Especially, Conway polyhedron notation is implemented. The notation can be observed as interactive animation.

***Keywords*—**Conway polyhedron notation, Polyhedral graph, Visualization.

I. Introduction

Conway polyhedron notation is used for describing various polyhedra based on Platonic solids as seed polyhedra [1-2]. The author has recently developed an interactive modeling system of polyhedra by means of graph drawing and simulated elasticity, mainly for educational purpose [10-13]. In this paper, we present an implementation of the full set of the notation on the system. Using this implementation, we can observe all of the operations of the notation as animation interactively.

II. Conway Polyhedron Notation

John Conway has proposed a notation for describing various polyhedra based on five Platonic solids as seed polyhedra. Table 1 shows the complete set of operations of the notation.

TABLE 1 The list of operators of Conway polyhedron notation.

Operator	Name	Description
d	dual	Each vertex becomes a new face and each face becomes a new vertex
t	truncate	Vertices are truncated
a	ambo	Vertices are truncated to the edge mid point
b	bevel	New faces are added in place of edges and vertices
e	expand	Each vertex makes a new face and each edge makes a new quadrangle
s	snub	Each vertex makes a new face and each edge makes a pair of triangles
k	kiss	Dual of truncate
j	join	Dual of ambo
m	meta	Dual of bevel
o	ortho	Dual of expand
g	gyro	Dual of snub

H. Nonaka is with Hokkaido University, Sapporo 060 0814, Japan (corresponding author to provide; e-mail: nonaka@main.ist.hokudai.ac.jp).

TABLE II The list of Platonic solids, where p, q, r are the number of vertices, edges, and faces, respectively.

Symbol	Name of polyhedron	p	q	r
P_{3^3}	Tetrahedron	4	6	4
P_{4^3}	Cube	8	12	6
P_{3^4}	Octahedron	6	12	8
P_{5^3}	Dodecahedron	20	30	12
P_{3^5}	Icosahedron	12	30	20

TABLE III The list of Archimedean solids, where p, q, r are the number of vertices, edges, and faces, respectively.

Symbol	Name of polyhedron	p	q	r
$A_{(3\cdot4)^2}$	Cuboctahedron	12	24	14
$A_{4\cdot6\cdot10}$	Great Rhombicosidodecahedron	120	180	62
$A_{4\cdot6\cdot8}$	Great Rhombicuboctahedron	48	72	26
$A_{(3\cdot5)^2}$	Icosidodecahedron	30	60	32
$A_{3\cdot4\cdot5\cdot4}$	Small Rhombicosidodecahedron	24	48	26
$A_{3\cdot4^3}$	Small Rhombicuboctahedron	24	60	38
$A_{3^4\cdot4}$	Snub Cube	24	36	14
$A_{3^4\cdot5}$	Snub Dodecahedron	60	150	92
$A_{3\cdot8^2}$	Truncated Cube	24	36	14
$A_{3\cdot10^2}$	Truncated Dodecahedron	60	90	32
$A_{5\cdot6^2}$	Truncated Icosahedron	60	90	32
$A_{4\cdot6^2}$	Truncated Octahedron	24	36	14
$A_{3\cdot6^2}$	Truncated Tetrahedron	12	18	8

Five Platonic solids and thirteen Archimedean solids are listed in Tables 2-3. For example, $A_{(3\cdot4)^2}$ indicates that two regular triangles and two squares are gathered alternately on each vertex. Archimedean solid can be expressed using Conway notation as follows:

$$A_{(3\cdot4)} = aP_{3^4} = aP_{4^3}\ ,\ \ A_{4\cdot6\cdot10} = bP_{3^5} = bP_{5^3}\ ,$$
$$A_{4\cdot6\cdot8} = bP_{3^4} = bP_{4^3}\ ,\ \ A_{(3\cdot5)^2} = aP_{3^5} = aP_{5^3}\ ,$$
$$A_{3\cdot4\cdot5\cdot4} = eP_{3^5} = eP_{5^3}\ ,\ \ A_{3\cdot4^3} = eP_{3^4} = eP_{4^3}\ ,$$
$$A_{3^4\cdot4} = sP_{3^4} = sP_{4^3}\ ,\ \ A_{3^4\cdot5} = sP_{3^5} = sP_{5^3}\ , A_{3\cdot8^2} = tP_{4^3}\ ,$$
$$A_{3\cdot10^2} = tP_{5^3}\ ,\ A_{5\cdot6^2} = tP_{3^5}\ ,\ A_{4\cdot6^2} = tP_{3^4}\ ,\ \ A_{3\cdot6^2} = tP_{3^3}\ ,$$
$$\text{and additionally, } P_{3^4} = aP_{3^3}\ ,\ P_{3^5} = sP_{3^3}\ . \tag{1}$$

III. Graph Operation for Polyhedra

The author previously presented an operation-based notation for Archimedean graph [12]. Three operations are defined for polyhedral graph: *edge contraction*, *vertex splitting*, and *diagonal addition* (Figure 1 – 3).

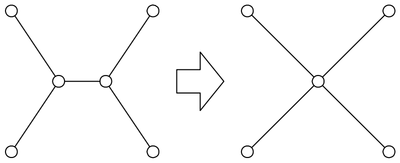


Fig. 1. An example of *edge contraction* operations.

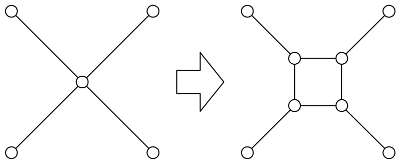


Fig. 2. An example of *vertex splitting* operations by the present definition in this paper.

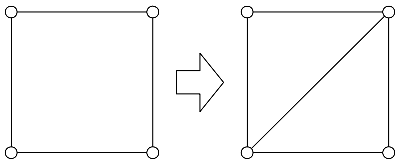


Fig. 3. An example of *diagonal addition* operations.

A polyhedral graph $G = \{V, E, F\}$ is defined by the set of vertices $V = \{v_0, \cdots, v_{p-1}\}$, edges $E = \{e_0, \cdots, e_{q-1}\}$, and faces $F = \{f_0, \cdots, f_{r-1}\}$. G is planar and 3-connected graph. The set F is subdivided as follows:

$$\left(F = \bigcup_{i=3,4,5,\cdots} F_i\right) \wedge \left(F_j \cap F_k = \phi \ \vee \ j = k\right), \qquad (2)$$

where F_n denotes the set of faces with n sides.

An *edge contraction* is a graph contraction of an edge (Figure 1). A *vertex splitting* is defined conventionally as the reverse of *edge contraction*, but in this paper, we define a *vertex splitting* of $v \in V$ as the composition of the operations of subdivision of incident edges on v, connecting the new vertices in a proper order, and deleting the vertex v (Figure 2). A *diagonal addition* to a face in F_4 is an edge addition between a pair of non-adjacent vertices in a quadrangular face (Figure 3).

The operation of *diagonal addition* is applied only when the graph G is obtained by applying "expand" to other graph H. Following statement holds for an arbitrary polyhedral graph G:

$$\begin{aligned}
&\forall G \exists H.(G = eH) \Leftrightarrow \forall G \exists \tilde{G}_0 \exists \tilde{G}_1 \forall u \forall v \exists w. \ [\\
&\quad (u \in \tilde{V} \Rightarrow u \in \tilde{V}_0 \vee u \in \tilde{V}_1) \wedge \\
&\quad (\neg\, u \in \tilde{V}_0 \vee \neg\, u \in \tilde{V}_1) \wedge \\
&\quad ((u,v) \in \tilde{E} \Rightarrow (u \in \tilde{V}_0 \wedge v \in \tilde{V}_1) \vee (u \in \tilde{V}_1 \wedge v \in \tilde{V}_0)) \wedge \\
&\quad (w \in \tilde{V}_0 \wedge \deg(w) = 3) \wedge \\
&\qquad (u \in \tilde{V}_1 \Rightarrow \deg(u) = 4)\ ],
\end{aligned} \qquad (3)$$

where $\tilde{G} = \{\tilde{V}, \tilde{E}, \tilde{F}\}$ is the dual of $G = \{V, E, F\}$. The first three clauses inside of the bracket mean that $\tilde{G}$ is a bipartite graph and $\tilde{V}$ is subdivided to $\tilde{V}_0$ and $\tilde{V}_1$. The fourth clause denotes that $\tilde{G}_0$ includes at least one vertex with degree 3, and G_0 contains at least one triangle. The last clause denotes that $\tilde{G}_1$ is a 4-regular graph, and all faces in G_1 are quadrangles, which will be applied *diagonal addition*. Quadrangles in G_0 should not be applied. Consequently, the operation of *diagonal addition* is performed by following algorithm:

1. Find a triangle *f* in G, and if not found, return false.
2. Search the dual graph $\tilde{G}$ from *f* in breadth-first way.
3. If the depth is odd number, in other words, if the distance (path length) from *f* is odd number, add a diagonal to *f*.
4. If not all the nodes in $\tilde{G}$ are traversed, go to step 2.
5. Return true.

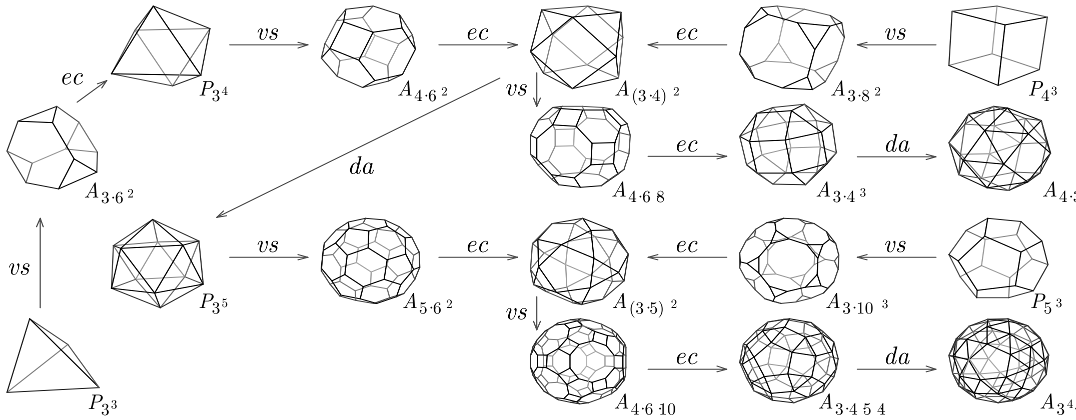


Fig. 4. Relations of 5 Platonic graphs and 13 Archimedean graphs induced by *edge contraction* (ec), *vertex splitting* (vs), and *diagonal addition* (da).

The operators of Conway notation can be expressed using graph operations for the isomorphic graph as follows. Truncate "t" is equivalent to *vertex splitting*. Ambo "a" is equivalent to the composition of *vertex splitting* and *edge contraction* for the original edges before applying *vertex splitting*. Bevel "b" is equivalent to "ta". Expand "e" is equal to two consecutive ambos "aa". Snub "s" is applying "e" followed by *diagonal addition*. The operators of the remainder are duals of above operators. Figure 4 shows the relations of 5 Platonic graphs and 13 Archimedean graphs induced by the three graph operations, and they correspond to the expressions in (1).

IV. Interactive Modeling System of Polyhedra

The interactive modeling system of polyhedron consists of three subsystems: graph input subsystem, wire-frame subsystem, and polygon subsystem [11].

Figure 5 shows a screen shot of graph input subsystem. The first step of the modeling of polyhedron is drawing a polyhedral graph isomorphic to the polyhedron. In the subsystem, vertex addition, vertex deletion, edge addition, and edge deletion are implemented as fundamental operations.

Figure 6 shows a screen shot of wire-frame subsystem. After constructing a polyhedral graph, the next step is arranging vertices in 3D space with virtual springs and Hooke's law. Wire-frame polyhedron can be formed by controlling the natural length of virtual spring corresponding to three types of binary relations between pairs of vertices.

Figure 7 shows a screen shot of polygon subsystem. After arranging vertices in 3D space, the last step is detecting faces, selecting appropriate faces, and rendering the solid. Detecting n-regular polygon is equivalent to finding simple closed path with length n.

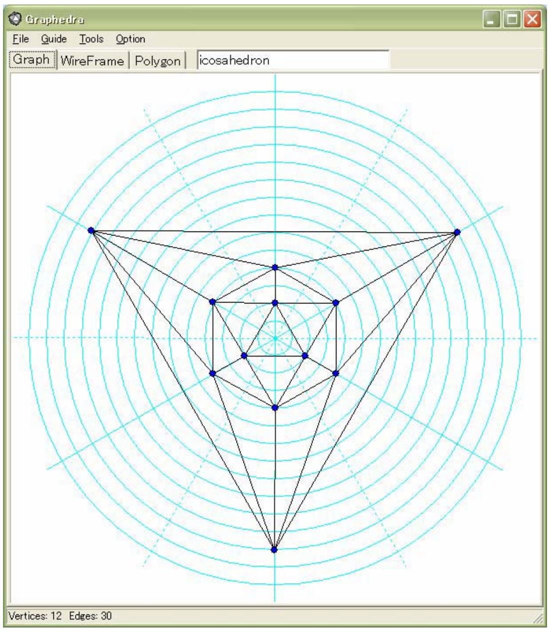


Fig. 5. Screen shot of graph input subsystem.

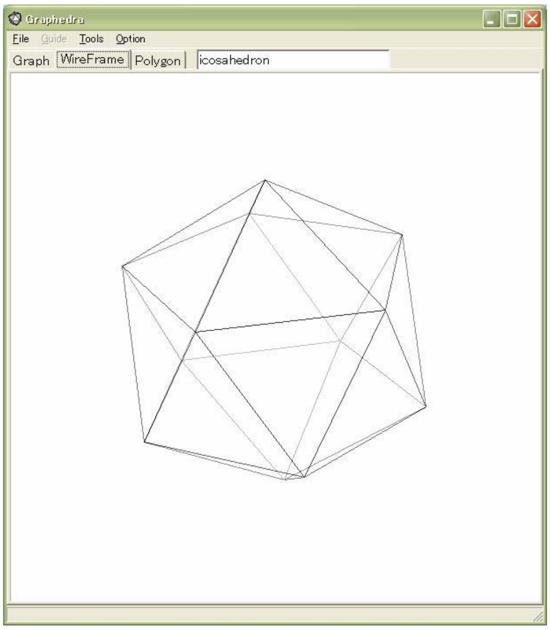


Fig. 6. Screen shot of wire-frame subsystem.

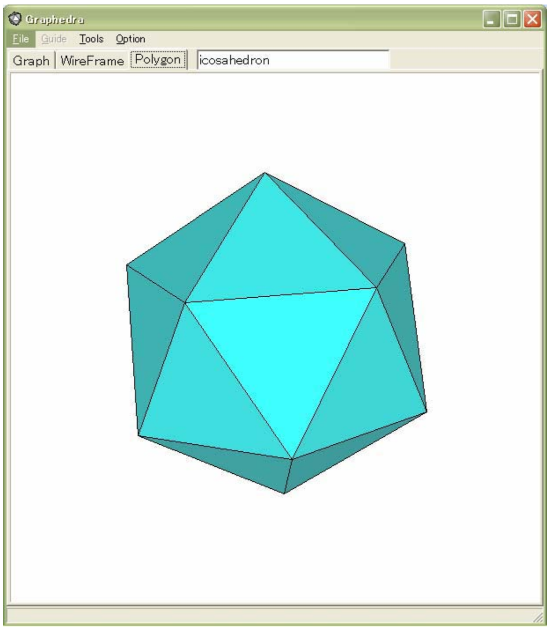


Fig. 7. Screen shot of polygon subsystem.

Figure 8 is a screen shot of the tool box for Conway polyhedron notation. The captions of buttons indicate the corresponding operations of Conway notation. These operations can be applied in both the wire-frame subsystem and the polygon subsystem.

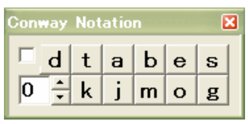
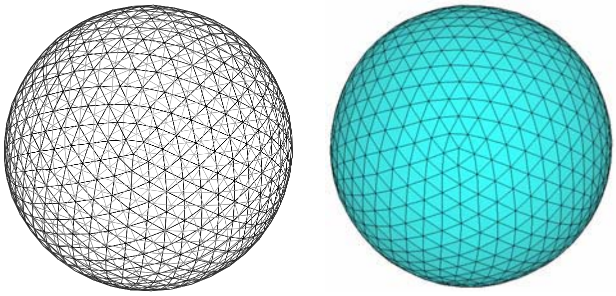


Fig. 8. Tool box for Conway polyhedron notation

By pressing a button on the tool box, corresponding operation is applied to the polyhedron with animation. The text box with spin buttons stands for an additional natural number n used with "t" and "k". If $n \geq 3$, "nt" means to "truncate" only vertices with degree n, and "nk" means to apply "kiss" only n-gons. Figure 9 shows a transition after pressing "s" for a icosahedron P_{3^5} . As a result, snub dodecahedron $A_{3^4\cdot 5}$ is obtained via several Archimedean solids: $A_{5\cdot 6^2}$, $A_{(3\cdot 5)^2}$, $A_{4\cdot 6\cdot 10}$, and $A_{3\cdot 4\cdot 5\cdot 4}$.

The polyhedra formed by the system are not restricted to semi-regular polyhedra or uniform polyhedra. For example, by pressing the buttons "t" and "d" four times alternately for an icosahedron, a series of polyhedra are displayed with animation one after another, and finally a polyhedron depicted in Figure 10 is obtained, however it is not a uniform polyhedron. It is expressed by Conway notation as $dtdtdtdtP_{3^5}$.



(a) Wire-frame (b) Polygon

Figure 10. The result of pressing "tdtdtdtd" for an icosahedron

REFERENCES

[1] G. W. Hart, "Conway Notation for Polyhedra", URL: http://www.gergehart.co/virtual-polyhedra/conway_notation.html.

[2] E. W. Weisstein, "Conway Polyhedron Notation", MathWorld.

[3] H. S. M. Coxeter, "Regular Polytopes", Dover Publications, 1973.

[4] P. Martin, and D. Martin, "PolyFormes: Software for the declarative modelling of polyhedra", The Visual Computer, 15 (2), pp. 55-76, 1999.

[5] G. W. Hart, and H. Picciotto, "Zome Geometry: Hands on Learning With Zome Models", Key Curriculum Pr., 2001.

[6] A. Holden, "Shapes, Space, and Symmetry", Columbia University Press, 1971

[7] T. Nishizeki and N. Chiba, "Planar Graphs – Theory and Algorithms", Dover Publications, 1988.

[8] T. Tyler, "Springie: A tensegrity simulator using Java, VRML and POV Ray", http://springie.com/.

[9] K. Inoue, K. Shimada, and K. Chilaka, "Solid Model Reconstruction of Wireframe CAD Models Based on Topological Embeddings of Planar Graphs", Journal of Mechanical Design, **125**, pp. 434-442, 2003.

[10] B. Grünbaum, "Graphs of polyhedra; polyhedra as graphs", Discrete Mathematics, **307**, pp. 445-463, 2007.

[11] H. Nonaka, "Detection of Faces in Wire-Frame Polyhedra", Proceedings of International Conference on Computer Graphics Theory and Applications, pp. 134-137, Jan. 2008.

[12] H. Nonaka, "Interactive Modeling of Uniform Polyhedra by Means of Graph Drawing", Proceedings of International Conference on Information Technology and Applications, 5, 822-827, 2008

[13] H. Nonaka, "Operation-Based Notation for Archimedean Graph", Proceedings of World Multi-Conference on Systemics, Cybernetics and Informatics, 12, 2008.

[14] H. Nonaka, "Modeling of Kepler-Poinsot Solid Using Isomorphic Polyhedral Graph", International Journal of Computer Science, 3 (1) 64-67, 2008, http://www.waset.org/ijcs/former.html

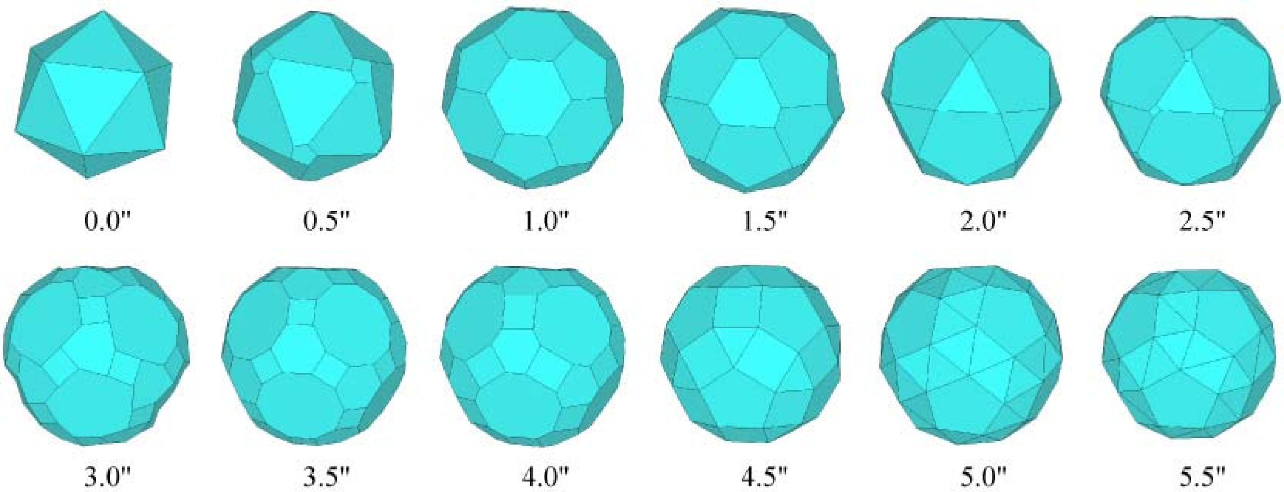


Fig. 9. Transition of polyhedron about every 0.5 seconds after applying "snub" for an icosahedron. During the animation, truncated icosahedron, icosidodecahedron, great rhombicosidodecahedron, small rhombicosidodecahedron, and finally, snub dodecahedron are observed.